# Applications of Computational Algebraic Topology

## Aplicaciones de la Topología Algebraica Computacional

**Julio Rubio**[1],[a]

**Abstract.** In this note a course given at the "UN Encuentro de Matemáticas 2016" held in Bogotá (Colombia) is described. The course was divided into four parts, devoted, respectively, to basic definitions in Algebraic Topology, to the algorithmic aspects of Topology, to the Kenzo computer system and, finally, to the applications of Computational Algebraic Topology. In the first talk the basic concepts of Algebraic Topology were presented, stressing the importance of replacing the topological space notion by other combinatorial analogues, easing the introduction of homological techniques. In the second talk, we reviewed several algorithms devised to compute homological objects. In the third talk, we described the Kenzo program, using its new version accessible via Internet. In the last talk, persistent homology was introduced and an actual application of topology to image processing was presented, in the frame of a project related to the fight against neurodegenerative diseases, as Alzheimer.

**Keywords:** Topology, Algorithmic, Programs, Applications.

**Resumen.** En este texto describimos un curso impartido en el congreso "UN Encuentro de Matemáticas 2016" que tuvo lugar en Bogotá (Colombia). El curso fue dividido en cuatro partes. En la primera se presentaron definiciones básicas de la Topología Algebraica. La segunda parte estuvo dedicada a los aspectos algorítmicos de la Topología, y en la tercera se habló del sistema de cálculo por computador Kenzo. La última parte trató sobre las aplicaciones de la Topología Algebraica Computacional. El curso comenzó recordando la importancia de reemplazar la noción de espacio topológico por otras combinatoriales, que facilitan la introducción de técnicas homológicas. Más adelante se

---

[1]Departamento de Matemáticas y Computación, Universidad de La Rioja (Spain), La Rioja, España
[a]julio.rubio@unirioja.es

presentaron algoritmos para calcular la homología, y se explicó cómo esos algoritmos están implementados en el sistema Kenzo. Por último, introdujimos la homología persistente, y explicamos una aplicación real de la topología para el procesamiento de imágenes digitales, en el marco de un proyecto dedicado a la lucha contra enfermedades neurodegenerativas, como el Alzheimer.

What is the relation between synapses in the brain and algebraic topology? Well, one could always adduce that in order to think of algebraic topology some synaptic activity is needed. But, seriously speaking, to find an answer to that question, we have to tell an exciting story of scientific collaboration. In 2011, Miguel Morales, leader of a team of biologists, met our Computer Science team, and explained to us that they were studying the effect of some specially-designed drugs in the evolution of synaptic density in the brain, and that this analysis could be instrumental in the fight against neurodegenerative diseases, as Alzheimer. Their particular problem was that they had a lot of pictures of neurones, taken by means of a confocal microscopy, and they had to count the number of synapses by hand. This task was very time-consuming, and not really related to the essential biological research. In addition, it was tainted by subjective factors. So, his specific question was: could you help us to automate this task? The material answer to that question was the plugin `SynapcountJ` which we developed some months later. Interestingly enough, the central algorithm was based on the computation of the number of connected components in digital images, a typical tool of *Computational Algebraic Topology*. Details of this program, and other subsequent projects undertaken together with Miguel Morales' team, can be read in [1] (in Spanish), where further references can also be found.

The way from abstract pure mathematics to a computer program useful in biomedicine belongs to a long and remarkable tradition. The story starts at the end of the XIX century, when Poincaré introduced in his *Analysis Situs* the modern Topology and, at the same time, the modern *Algebraic Topology* through the notion of *fundamental group*. In the course, we quickly mentioned the historical birth of the main concepts. Topological spaces, as defined in General Topology textbooks, can produce extremely wild and badly-behaved spaces. So, a first step towards rendering topology more algebraic is to find combinatorial analogues of topological spaces. The path presented was:

1. Topological spaces;

2. CW-complexes;

3. Triangularizations;

4. Simplicial complexes;

5. Simplicial sets.

Once simplicial sets have been chosen as combinatorial models for topological spaces, the next step is to get algebraic structures from the geometric ones. The technique is to use the free Abelian group generated by a set. Taking profit of the graded structure inherent to simplicial sets (degree is identified with the geometrical dimension), we obtain a graded group associated to a simplicial set. Faces in each simplex can be composed formally to endow the graded group of a simplicial set with a structure of *chain complex*.

A *chain complex* $C_* = (C_p, d_p)$ is a collection of Abelian groups $(C_p)$, one for each $p \in \mathbb{Z}$, together with a homomorphism $d_p : C_p \to C_{p-1}$, such that, for all $p$, $d_{p-1} \circ d_p = 0$.

$$\cdots \leftarrow C_{p-1} \xleftarrow{d_p} C_p \xleftarrow{d_{p+1}} C_{p+1} \leftarrow \cdots .$$

The fundamental property of a chain complex is $d_p \circ d_{p+1} = 0$ implies $im\, d_{p+1} \subseteq \ker d_p$. Then the *p-homology group* of $C_*$, denoted by $H_p(C_*)$, is defined as the quotient $\ker d_p / im\, d_{p+1}$. This is one of the essential notions in Algebraic Topology and has many applications. In particular, $H_0(C_*)$ measures the number of connected components of the initial space.

Once the homology groups were introduced, the main algorithm to compute them was described in the second talk: it is based on a diagonalization process of integer matrices known as *Smith Normal Form*. If the space to be studied has finitely many simplices in each dimension, then its homology groups are finitely generated Abelian groups. The arrows in the chain complex can be represented as integer matrices, and then the rang and torsion coefficients of the homology groups can be read directly on the diagonal matrices produced by the Smith Normal Form algorithm.

In advanced Algebraic Topology, when trying to compute *homotopy* groups of spaces of finite type (a problem much more difficult than computing homology groups), infinite dimensional spaces appear. The typical case is that of *loop spaces*. Iterated loop spaces are highly infinite, but, under good circumstances, their homology groups are finitely generated, so computable, in principle. There are specific algorithms to compute such homology groups, requiring Sergeraert's *effective homology*.

Let us comment with some detail on the notion of effective homology. Roughly speaking, the idea is to keep an explicit link among a big (frequently of infinite type) chain complex and a small (of finite type) one, in such a way that the homology of the big chain complex can be obtained through that of the small one. The mentioned explicit link is organised by means of the (chain) morphism notion.

A *morphism* $f = (f_p)$, of degree $k$, from a chain complex $(C_p, d_p)$ to another $(C'_p, d'_p)$ is a collection of homomorphisms

$$f_p : C_p \to C'_{p+k}.$$

This is expressed by the following diagram, generally **not assumed commutative**

$$\cdots \quad \leftarrow \quad C_{p-1} \quad \overset{d_p}{\leftarrow} \quad C_p \quad \overset{d_{p+1}}{\leftarrow} \quad C_{p+1} \quad \leftarrow \quad \cdots$$

$$f_{p-1} \downarrow \qquad\qquad f_p \downarrow \qquad\qquad f_{p+1} \downarrow$$

$$\cdots \quad \leftarrow \quad C'_{p+k-1} \quad \overset{d'_{p+k}}{\leftarrow} \quad C'_{p+k} \quad \overset{d'_{p+k+1}}{\leftarrow} \quad C'_{p+k+1} \quad \leftarrow \quad \cdots$$

Then, the structure composed of two chain complexes and the corresponding link is called by Sergeraert a *reduction*.

A *reduction* is a 5–tuple $(\hat{C}, C, f, g, h)$:

$$\hat{C} \overset{h}{\longrightarrow} {}^s\hat{C}$$
$$f \downarrow\uparrow g$$
$$C$$

where $\hat{C}$ and $C$ are chain complexes, $f$ and $g$ chain complex morphisms and $h$ a homotopy operator. Hereafter, $\hat{C}$ is called the *top chain complex* and $C$ the *bottom chain complex*. ${}^s\hat{C}$ is $\hat{C}$ shifted, i.e. $h$ has degree 1. The mappings $f$, $g$, $h$, together with the differential operator $d$ on $\hat{C}$, must verify the following relations:

$$f \circ g = 1_C$$
$$h \circ d + d \circ h = 1_{\hat{C}} - g \circ f$$
$$f \circ h = 0$$
$$h \circ g = 0$$
$$h \circ h = 0$$

The morphisms $f$ and $g$ and the homotopy operator $h$ describe the (big) chain complex $\hat{C}$ as the direct sum

$$\hat{C} = \hat{C}_1 \oplus \hat{C}_2$$

where $\hat{C}_1 = Im(g) \simeq C$ and $\hat{C}_2 = Ker(f)$ ($\hat{C}_2$ is acyclic).

$$\hat{C} \overset{h}{\longrightarrow} {}^s\hat{C}$$
$$f \downarrow\uparrow g$$
$$C$$

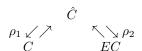The morphisms $f$ and $g$ and the homotopy operator $h$ describe the (big) chain complex $\hat{C}$ as the direct sum

$$\hat{C} = \hat{C}_1 \oplus \hat{C}_2$$

where $\hat{C}_1 = Im(g) \simeq C$ and $\hat{C}_2 = Ker(f)$ ($\hat{C}_2$ is acyclic).

A *homotopy equivalence* between two chain complexes $C$ and $EC$ is a pair of reductions:

$$\begin{array}{ccc} & \hat{C} & \\ \rho_1 \swarrow \nearrow & & \nwarrow \searrow \rho_2 \\ C & & EC \end{array}$$

If $C$ and $EC$ are *free* chain complexes, a usual chain equivalence between them can thus be organized. Frequently the chain complexes $C$ and $\hat{C}$ are of infinite type and on the contrary, the chain complex $EC$ is of finite type, so that $EC$ can be understood as a description of the homology of $C$. More precisely, $EC$ is a tool allowing the computation of the homology of $C$. The chain complex $\hat{C}$ is only an intermediate object. When $EC$ is an effective chain complex, we call the whole structure an *object with effective homology.*

The effective homology method has been implemented in the software system called *Kenzo*, available at `https://www-fourier.ujf-grenoble.fr/~sergerar/`. Kenzo is one of the computer programs devoted to calculate in Algebraic Topology. It is unique in dealing with infinitely generated spaces, and it has been used to perform specific computations in applications. It also includes packages to compute persistent homology, and to use discrete vector fields to reduce the size of the matrices involved in the determination of the homology of digital images. This last technique has been instrumental to some of the applications in the field of neuronal analysis as it is described in [1] and with more detail in the literature references of that paper.

# References

[1] A. Romero, J. Heras, G. Mata, M. Morales and J. Rubio, *Procesamiento topo-geométrico de imágenes neuronales*, La Gaceta de La Real Sociedad Matemática Española **17** (2014), no. 1, 109–128.

[2] J. Heras, V. Pascual, J. Rubio y F. Sergeraert, *fKenzo: a user interface for computations in Algebraic Topology*, J. Symbolic Comput. **46** (2011), 685–698.

[3] ———, *fKenzo: una interfaz de usuario para realizar cálculos en Topología Algebraica*, Gac. R. Soc. Mat. Esp. **14** (2011), 205–308.

[4] M. Morales et al., *Sitio web del grupo*, `http://www.spineup.es`.

[5] A. Romero, G. Ellis y J. Rubio, *Interoperating between Computer Algebra systems: computing homology of groups with Kenzo and GAP*, Proceedings International Symposium on Symbolic and Algebraic Computation, 303–310, *ACM*, 2009.

[6] A. Romero, J. Heras, G. Mata, J. Rubio y F. Sergeraert, *Spectral sequences for computing persistent homology of digital images*, Conference on Applications of Computer Algebra (Málaga, 2013), 331–335.

[7] A. Romero y J. Rubio, *Computing the homology of groups: The geometric way*, J. Symbolic Comput. **47** (2012), 752–770.

[8] _____, *Homotopy groups of suspended classifying spaces: an experimental approach*, Math. Comp. **82** (2013), 2237–2244.

[9] A. Romero, J. Rubio y F. Sergeraert, *Computing Spectral Sequences*, J. Symbolic Comput. **41** (2006), 1059–1079.

[10] J. Rubio, *Programmation du Lemme de Perturbation*, Ann. Univ. Ferrara Sez. VII Sci. Mat. **38** (1992), 41–47.

[11] _____, *Integrating functional programming and symbolic computation*, Math. Comput. Simulation **44** (1997), 505–511.

[12] _____, *Topología Algebraica Efectiva*, Gac. R. Soc. Mat. Esp. **1** (1998), 198–208.

[13] _____, *On the role of formalization in Computational Mathematics*, Bull. Belg. Math. Soc. Simon Stevin **20** (2013), 385–403.

[14] J. Rubio y F. Sergeraert, *Homologie effective et suites spectrales d'Eilenberg-Moore*, C. R. Math. Acad. Sci. Paris **306** (1988), 723–726.

[15] _____, *Supports acycliques et algorithmique*, Astérisque **192** (1990), 35–55.

[16] _____, *Constructive Algebraic Topology*, Bull. Sci. Math. **126** (2002), 389–412.

[17] _____, *Postnikov "invariants" in 2004*, Georgian Math. J. **12** (2005), 139–155.

[18] _____, *Algebraic models for homotopy types*, Homology, Homotopy Appl. **7** (2005), 139–160.

[19] _____, *Computing with locally effective matrices*, Int. J. Comput. Math. **82** (2005), 1177–1189.